<u>CLAIM AMENDMENTS</u>

1.   (Currently Amended) A method comprising:

<u>selectively loading either a first module of the basic input/output system or a second module of the basic input/output system based on a system state that indicates a connection to a network;</u>

executing <u>said</u> [[a]] first basic input/output system module; and

dynamically linking to <u>said</u> [[a]] second basic input/output system module.

2.   (Original) The method of claim 1 further comprising:

storing said first module of a basic input/output system for a processor-based system on a first storage device prior to execution;

storing said second module of the basic input/output system on a second storage device prior to execution; and

enabling said second module to be executed conditionally depending on a state of said processor-based system.

3.   (Original) The method of claim 2 wherein storing said second module includes storing said second module in a storage associated with a network server accessible to said processor-based system over a network.

4.   (Original) The method of claim 1 further including detecting said system state during the boot sequence.

5.   (Original) The method of claim 4 including detecting whether or not the system is connected to a network during the boot operation.

6.   (Original) The method of claim 1 including dynamically linking to one of a plurality of modules, and exporting an offset to an entry point in one module to another module.

7.   (Original) The method of claim 6 including storing a secondary entry point in a module to locate a function within the module.

8.   (Original) The method of claim 7 including developing a segment address for said second module at run time.

9.   (Original) The method of claim 8 including providing a descriptor table which indicates a segment address for said second module.

10.   (Currently Amended) An article comprising a medium for storing instructions that cause a processor-based system to:

        selectively load either a first module of the basic input/output system or a second module of the basic input/output system based on a system state that indicates a connection to a network;

        execute [[a]] said first basic input/output system module; and

        dynamically link to [[a]] said second basic input/output system module.

11.   (Original) The article of claim 10 further storing instructions that cause a processor-based system to:

        access said first module of a basic input/output system on a first storage device;

        access said second module of the basic input/output system on a second storage device; and

        execute said second module conditionally depending on the state of said processor-based system.

12.   (Original) The article of claim 11 further storing instructions that cause a processor-based system to access said second module in a storage associated with a network server accessible to said processor-based system over a network.

13. (Original) The article of claim 11 further storing instructions that cause a processor-based system to execute said second module conditionally depending on whether or not the processor-based system is coupled to a network.

14. (Original) The article of claim 11 further storing instructions that cause a processor-based system to selectively access either a first module setting forth a first authentication protocol in a first storage device or a second module setting forth a second authentication protocol in a second storage device.

15. (Original) The article of claim 11 further storing instructions that cause a processor-based system to dynamically link said first and second modules.

16. (Original) The article of claim 11 further storing instructions that cause a processor-based system to detect said system state during the boot sequence.

17. (Original) The article of claim 16 further storing instructions that cause a processor-based system to detect whether the system is connected to a network during the boot operation.

18. (Original) The article of claim 11 further storing instructions that cause a processor-based system to dynamically link to one of a plurality of modules using offsets to entry points in said modules.

19. (Original) The article of claim 18 further storing instructions that cause a processor-based system to store a secondary entry point in a module to locate a function within the module.

20. (Original) The article of claim 19 further storing instructions that cause a processor-based system to develop a segment address for said second module at run time.

21.   (Original) The article of claim 20 further storing instructions that cause a processor-based system to provide a descriptor table which identifies the segment address for said second module.

22.   (Currently Amended) A processor-based system comprising:

a processor;

a first basic input/output system module executable by said processor; and

a second basic input/output system module executable by said processor, said second module being dynamically linked to said first module after selectively loading either said first module of the basic input/output system or said second module of the basic input/output system based on a system state that indicates a connection to a network..

23.   (Original) The system of claim 22 including a detector that detects a system state to determine whether said processor executes said second module.

24.   (Original) The system of claim 22 including a first storage for said first module and a second storage for said second module, said second storage being coupled to said processor-based system over a network.

25.   (Original) The system of claim 24 wherein said detector detects information about network access.

26.   (Original) The system of claim 25 wherein said first and second modules include different authentication protocols.

27.   (Original) The system of claim 26 wherein said processor executes said basic input/output system module on said second storage to implement a network authentication protocol.

28.   (Original) The system of claim 22 wherein said first module dynamically links to said second module, using an offset exported from said second module.

29.   (Original) The system of claim 28 wherein said first module uses a secondary entry point to locate a function in said second module.

30.   (Original) The system of claim 22 wherein said processor provides a descriptor table which includes a segment address for said second module.